



**Barcelona
Supercomputing
Center**
Centro Nacional de Supercomputación



Reproducible science at large scale within a continuous delivery pipeline: the BSC vision

Miguel Castrillo

BSC-ES Computational Earth Sciences

15/10/2019

ECMWF workshop: Building reproducible workflows for Earth Sciences

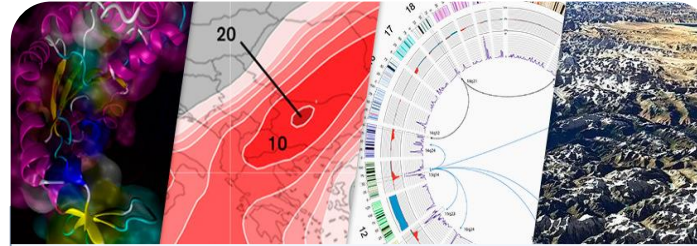
Barcelona Supercomputing Center

Centro Nacional de Supercomputación

BSC-CNS objectives



Supercomputing services
to Spanish and EU researchers



R&D in Computer, Life, Earth and
Engineering Sciences



PhD programme, technology
transfer, public engagement

BSC-CNS is
a consortium
that includes

Spanish Government

60%



Catalan Government

30%



Univ. Politècnica de Catalunya (UPC)

10%



MareNostrum 4

Total peak performance: **13,7 Pflops**

General Purpose Cluster:	11.15 Pflops	(1.07.2017)
CTE1-P9+Volta:	1.57 Pflops	(1.03.2018)
CTE2-AMD:	0.52 Pflops	(1.11.2019)
CTE3-Arm V8:	0.5 Pflops	(????)



Access: prace-ri.eu/hpc_acces



RED ESPAÑOLA DE
SUPERCOMPUTACIÓN

Access: bsc.es/res-intranet



Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

MareNostrum 1

2004 – 42,3 Tflops

1st Europe / 4th World

New technologies

MareNostrum 2

2006 – 94,2 Tflops

1st Europe / 5th World

New technologies

MareNostrum 3

2012 – 1,1 Pflops

12th Europe / 36th World

MareNostrum 4

2017 – 11,1 Pflops

2nd Europe / 13th World

New technologies

MareNostrum 5. A European pre-exascale supercomputer

- **200 Petaflops** peak performance (200×10^{15})
- **Experimental platform** to create supercomputing technologies “made in Europe”
- **223 M€** of investment



Hosting Consortium:

Spain Portugal Turkey Croatia



Mission of BSC Scientific Departments



Computer Sciences

To influence the way machines are built, programmed and used: programming models, performance tools, Big Data, computer architecture, energy efficiency



Earth Sciences

To develop and implement global and regional state-of-the-art models for short-term air quality forecast and long-term climate applications



Life Sciences

To understand living organisms by means of theoretical and computational methods (molecular modeling, genomics, proteomics)

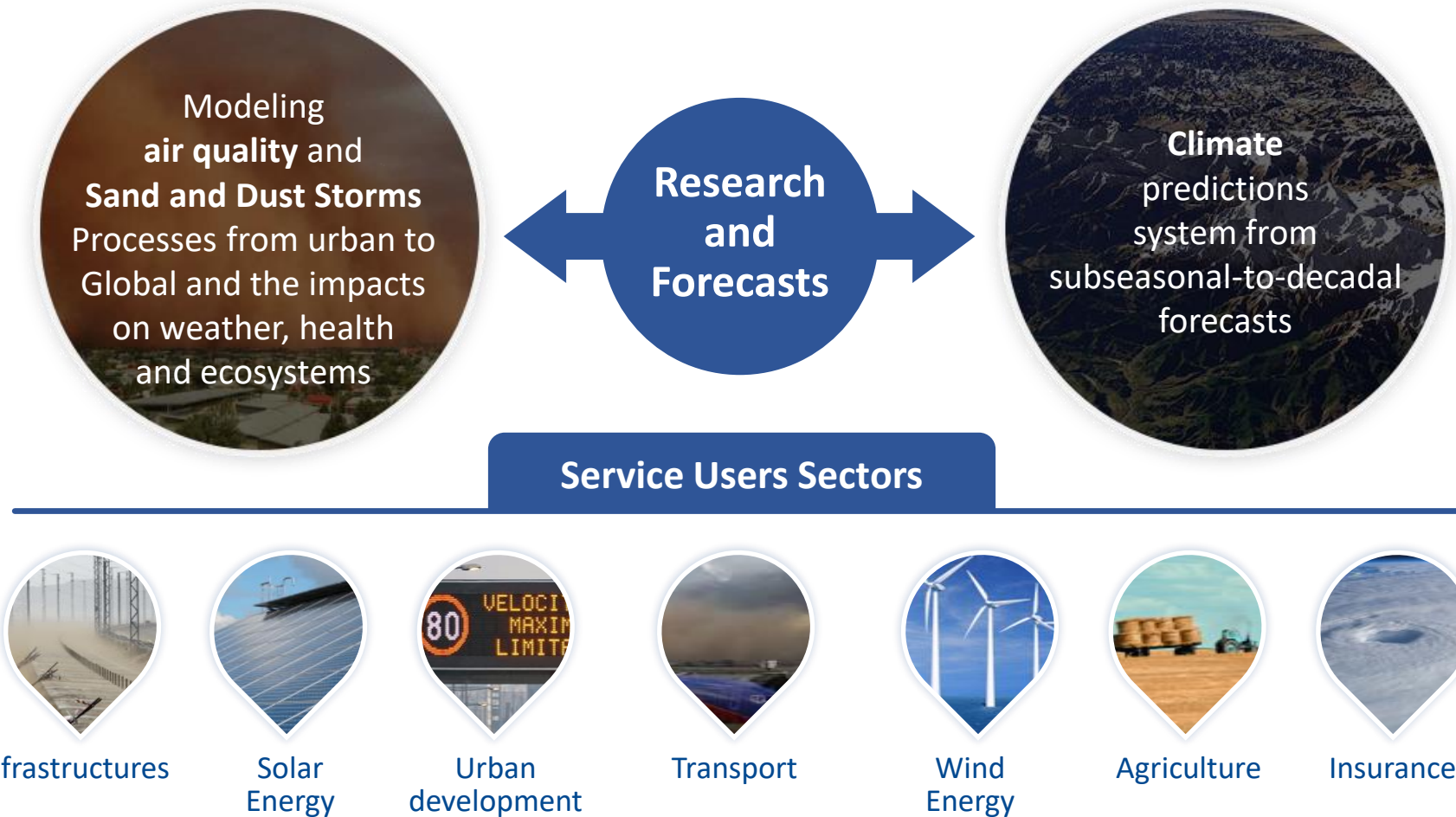


CASE

To develop scientific and engineering software to efficiently exploit super-computing capabilities (biomedical, geophysics, atmospheric, energy, social and economic simulations)

Earth Sciences

Environmental modelling and forecasting, with a particular focus on weather, climate and air quality



The Models & Workflows team

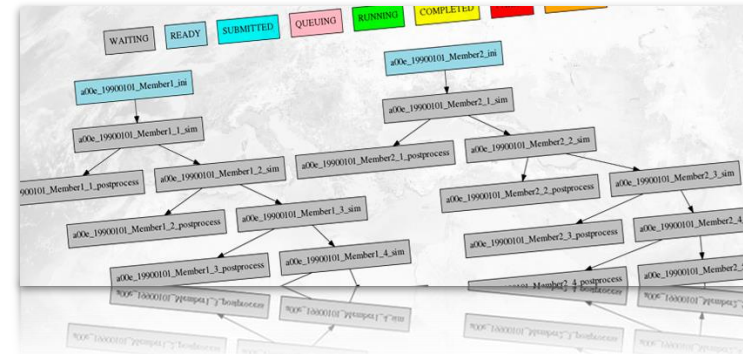
Models

Design, development, and deployment of Earth science **models** in close **collaboration** with the **scientific** groups aiming to understand and better predict the behaviour of Earth systems.

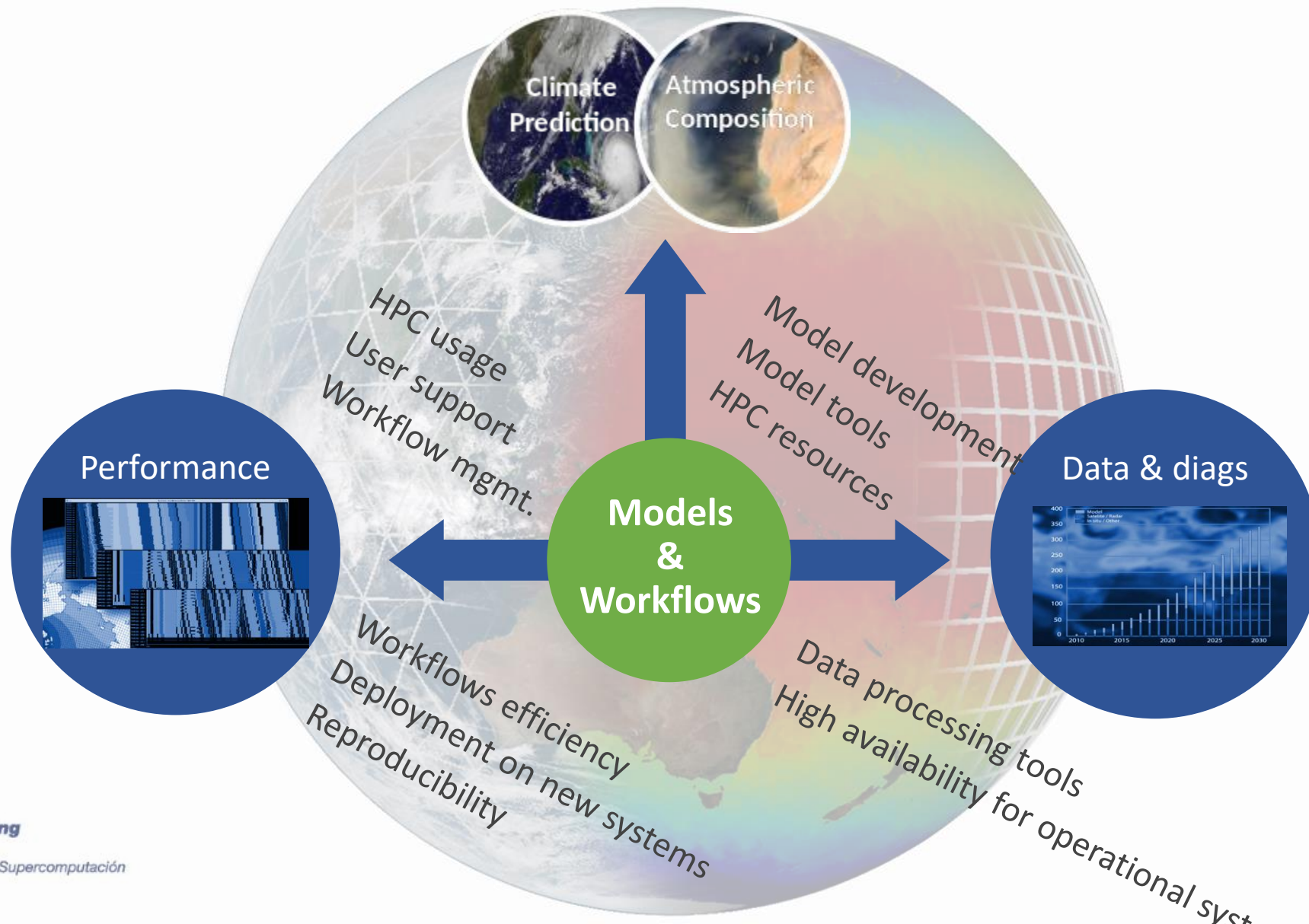
```
if not core_title:
    core_title = "CORE-3000"
else:
    core_title = "Scalability"
color_map = {"red": "red", "green": "green", "cyan": "cyan", "magenta": "magenta", "goldenrod": "goldenrod", "steelblue": "steelblue", "brown": "brown", "greenyellow": "greenyellow", "orange": "orange", "black": "black", "lavender": "lavender", "darkslategray": "darkslategray", "gray": "gray", "darkgray": "darkgray"}
bar_width = 50
for i in range(len(stage_list)):
    if speed:
        # Use mean
        # Show points for every element in all the series
        core_unique = sorted(core_series)
        ax.scatter([val for val in sorted(core_unique, reverse=True) if use_size else False], [color_map[stage_list[i]]], [stage_list[i]] * 40)
        if annotate:
            # Use size
            core_unique_list = sorted([core_unique, reverse=True])
            for j, core in enumerate(core_unique_list):
                for k, val in enumerate(core_series):
                    if core_series[i] == core:
                        ax.annotate(f'{core}_{i}_{j}', [core_unique_list[i], core_series[i]], [stage_list[i]] * 40)
            else:
                core_unique_list = sorted([core_unique, reverse=True])
                for j, core in enumerate(core_unique_list):
                    for k, val in enumerate(core_series):
                        if core_series[i] == core:
                            ax.annotate(f'{core}_{i}_{j}', [core_unique_list[i], core_series[i]], [stage_list[i]] * 40)
    else:
        # Use plot
        ax.scatter([val for val in core_series[i]], [color_map[stage_list[i]]], [stage_list[i]] * 40)
        ax.annotate(f'{core}_{i}_{j}', [core_unique_list[i], core_series[i]], [stage_list[i]] * 40)
    if not data:
        # Show the current series
        if not data:
            # Show the current series
            # Use size
            core_unique_list = sorted([core_unique, reverse=True])
            for j, core in enumerate(core_unique_list):
                for k, val in enumerate(core_series):
                    if core_series[i] == core:
                        ax.annotate(f'{core}_{i}_{j}', [core_unique_list[i], core_series[i]], [stage_list[i]] * 40)
            else:
                core_unique_list = sorted([core_unique, reverse=True])
                for j, core in enumerate(core_unique_list):
                    for k, val in enumerate(core_series):
                        if core_series[i] == core:
                            ax.annotate(f'{core}_{i}_{j}', [core_unique_list[i], core_series[i]], [stage_list[i]] * 40)
```

Workflows

Research and development of **methodologies** and **tools** that allow the **running** of scientific models in **production** and taking advantage of the increasing availability and variety of computing resources.



MWT synergies

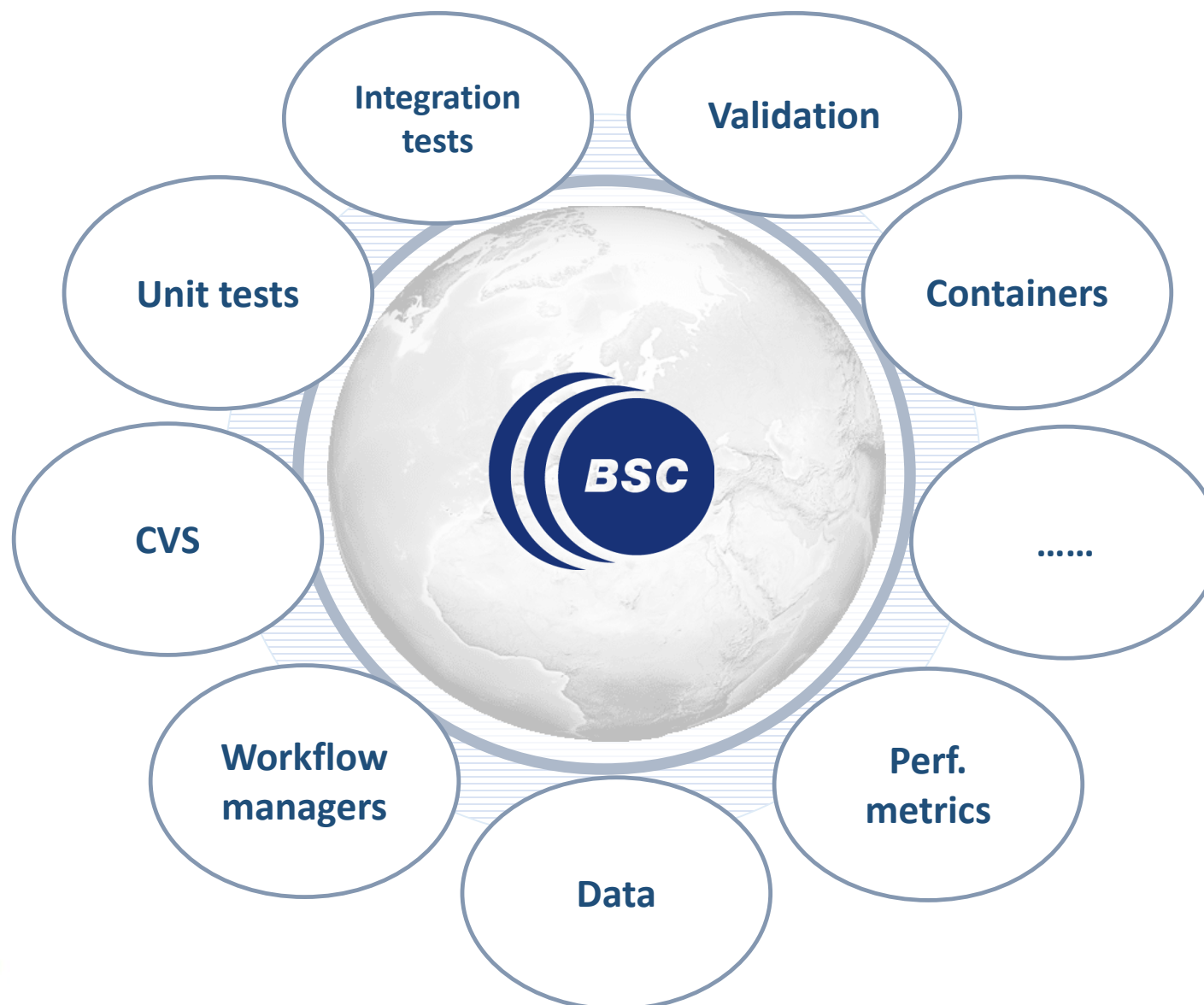


High Performance Computing in Earth Sciences

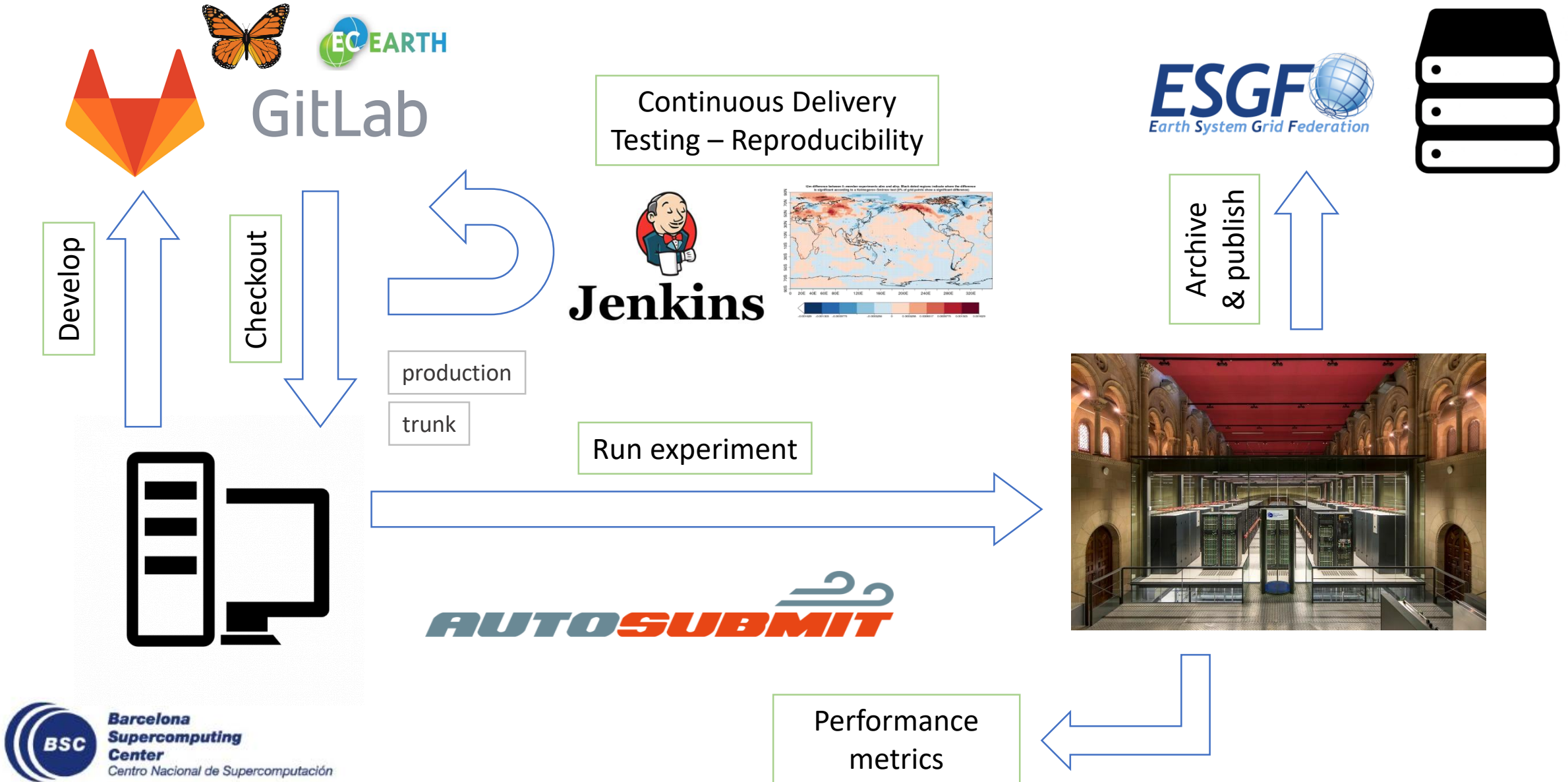
- Earth System Models (ESMs) are sophisticated tools with continuously increasing complexity:
 - More components of Earth System are included
 - Finer Spatial and Temporal resolutions
- This increase in complexity could be developed thanks to the important parallel advances in HPC



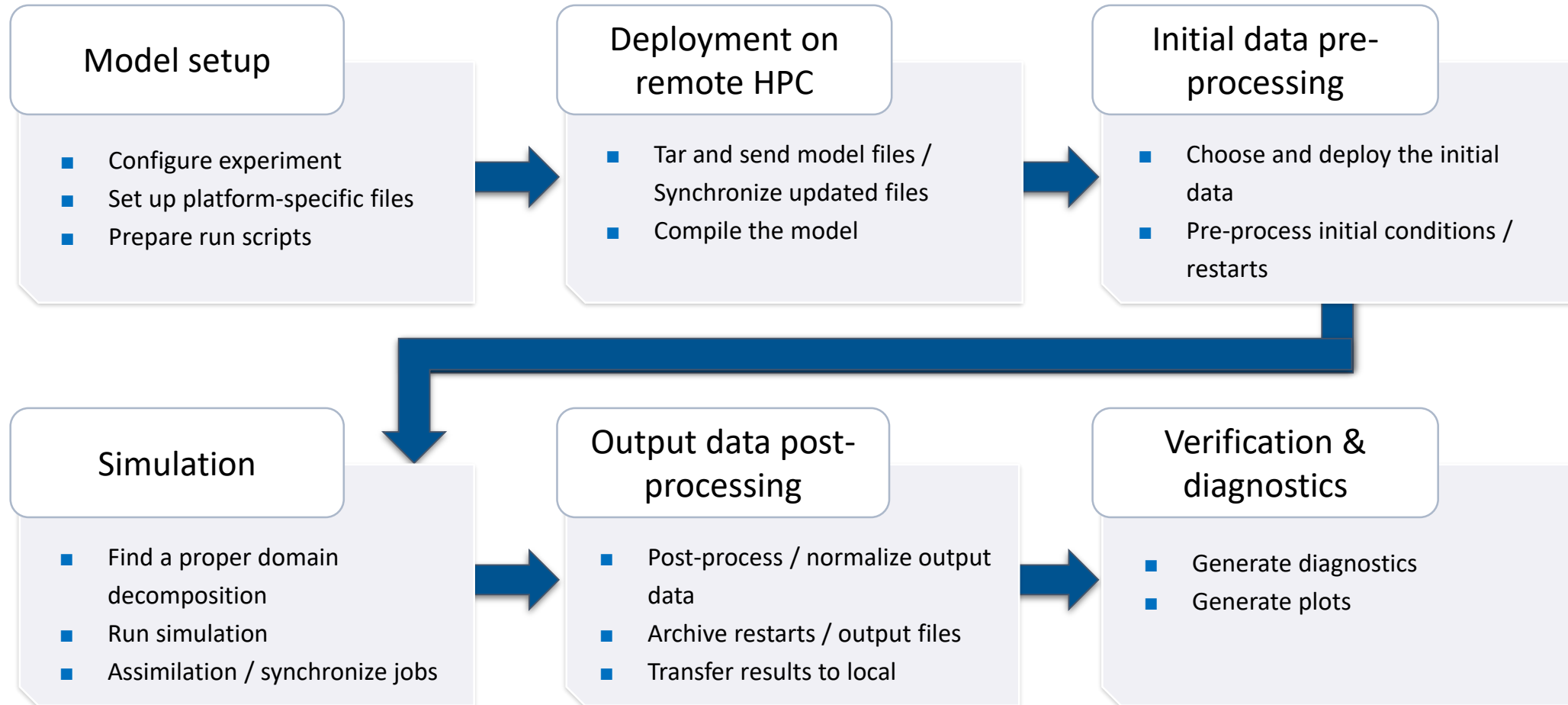
Workflows at BSC-ES



Our workflow



A workflow for ESM experiments



Workflow managers: motivation

Workflow managers are **essential** to carry out production experiments in an **efficient** way

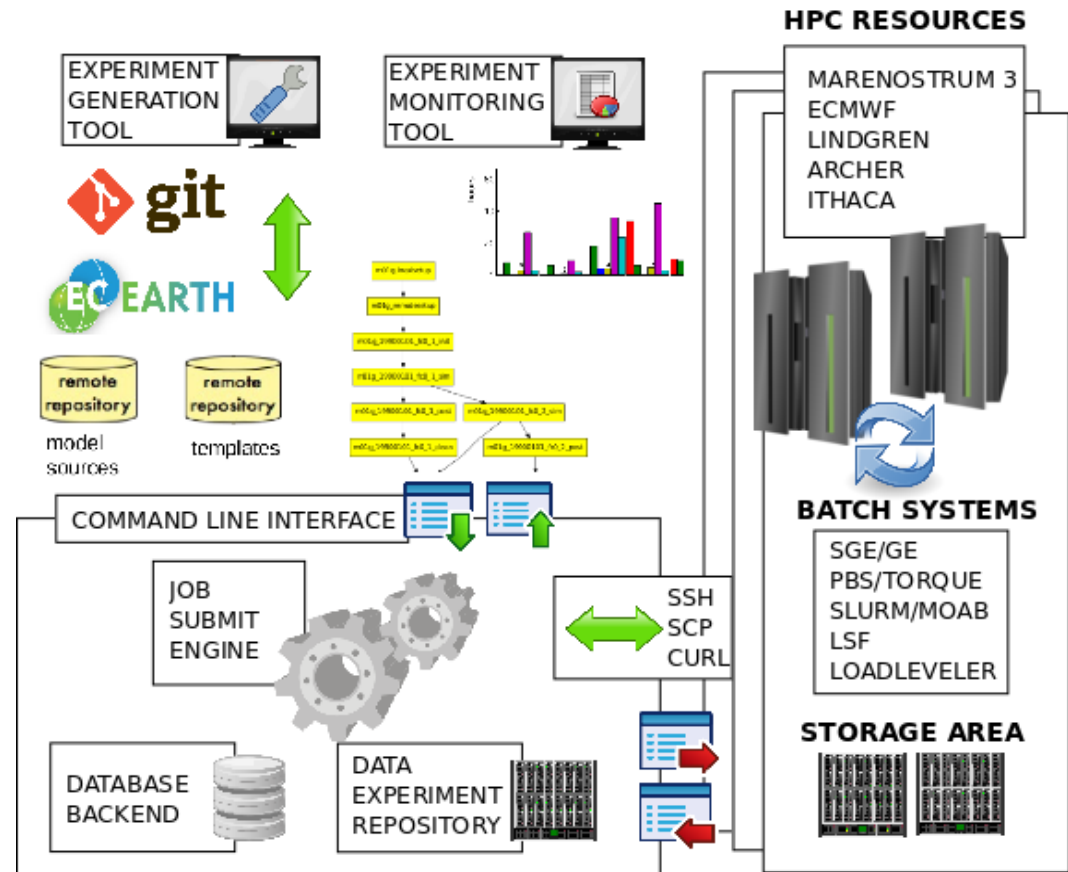
- Deal with workflow **complexity**
- Ensure **robustness & portability**
- **Usability** → Scientists more productive



Autosubmit

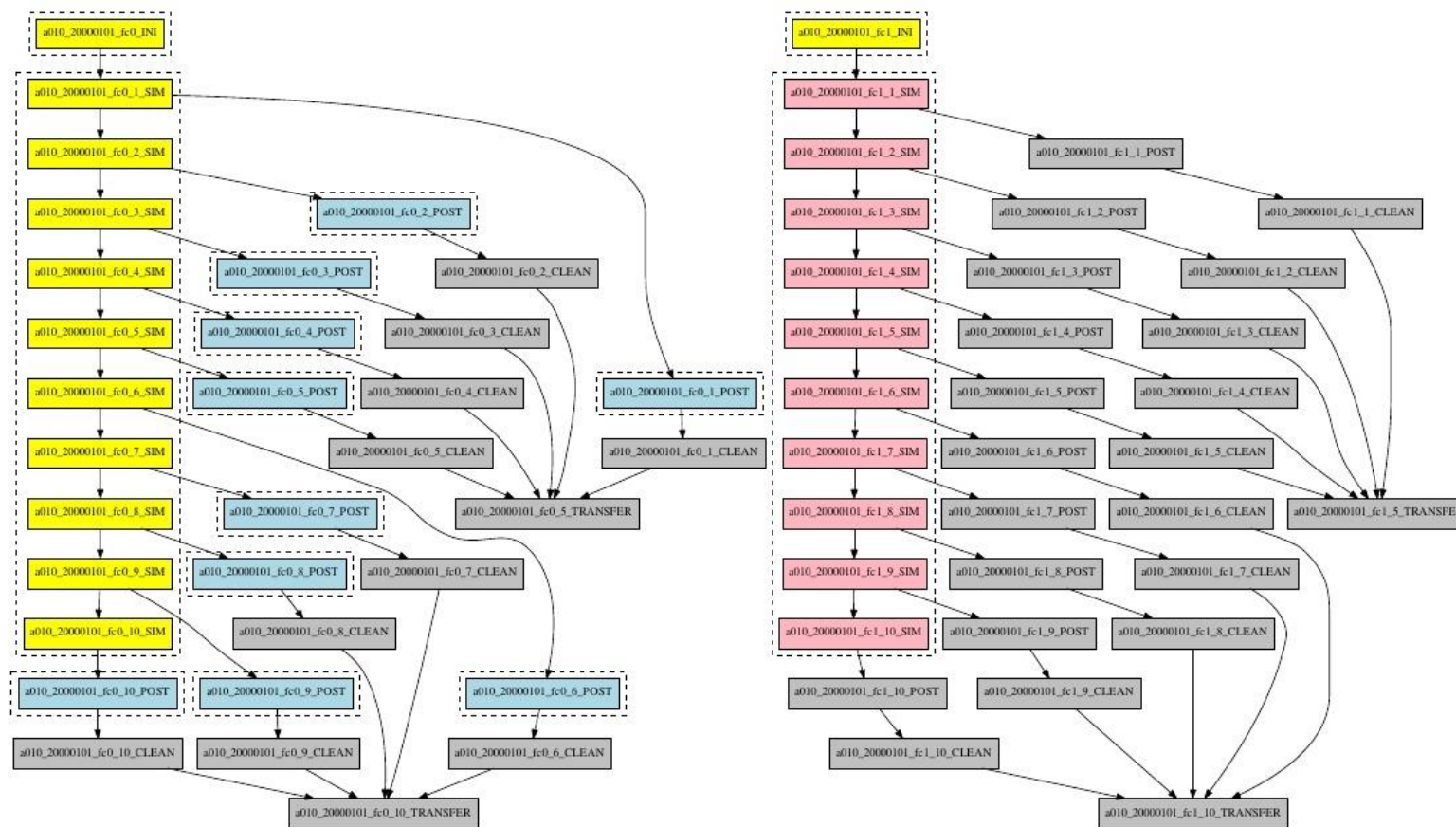
A **versatile** tool to manage Weather
and Climate Experiments in diverse
Supercomputing Environments:

<https://pypi.python.org/pypi/autosubmit>



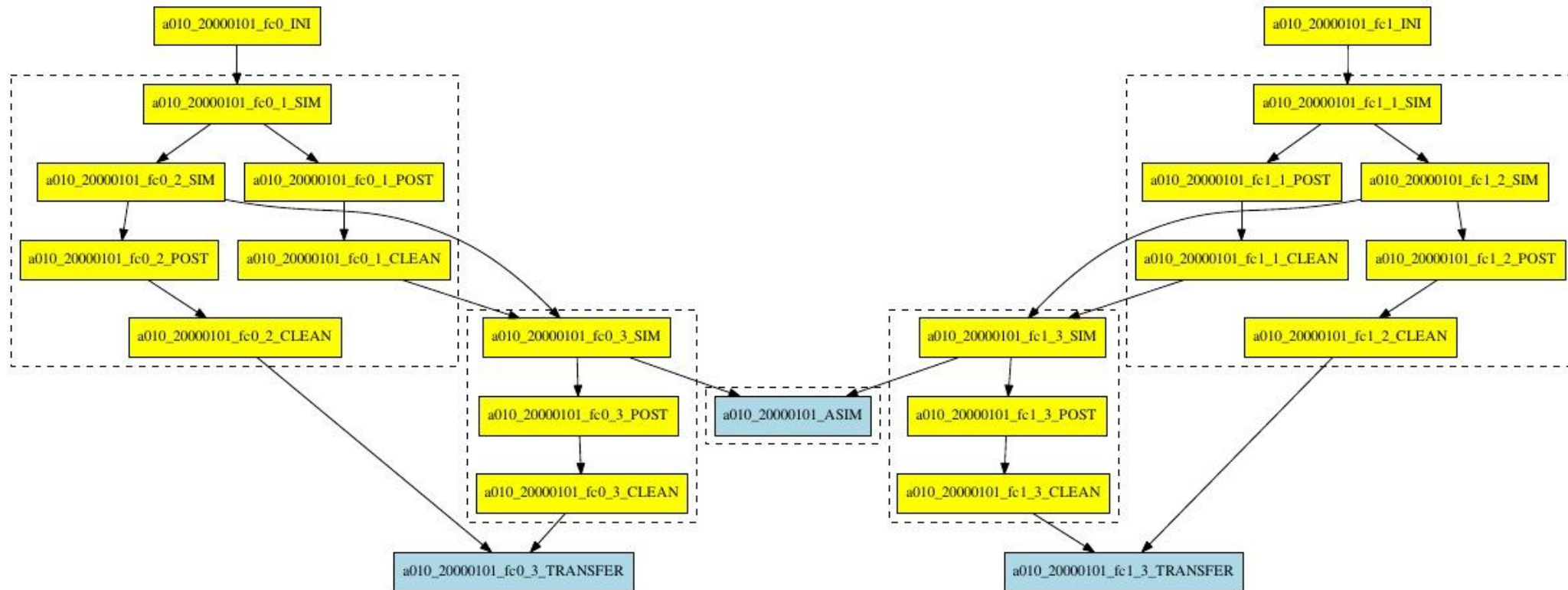
Autosubmit wrapper

Reducing queueing times by wrapping different jobs together



Autosubmit wrappers

Hybrid wrappers

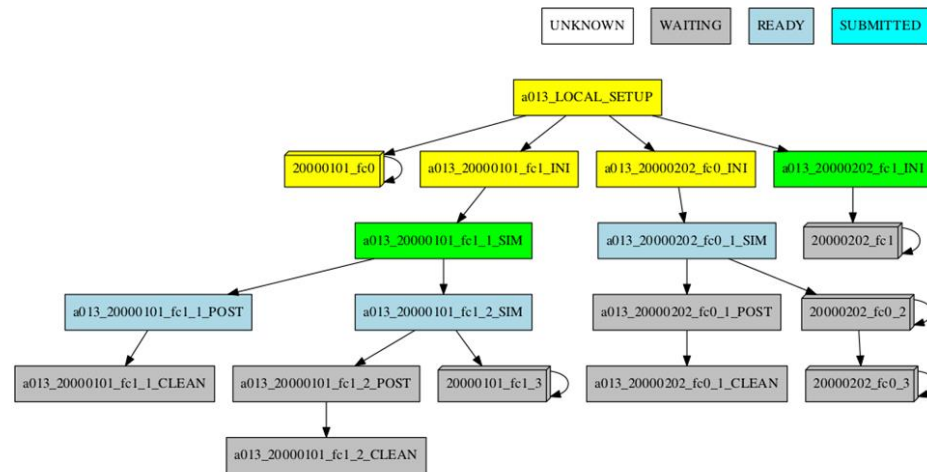


Autosubmit: lessons learned

- Workflows are getting more and more **complex**
- Workflow **managers** are required to **improve** in order to **deal** with this **complexity**

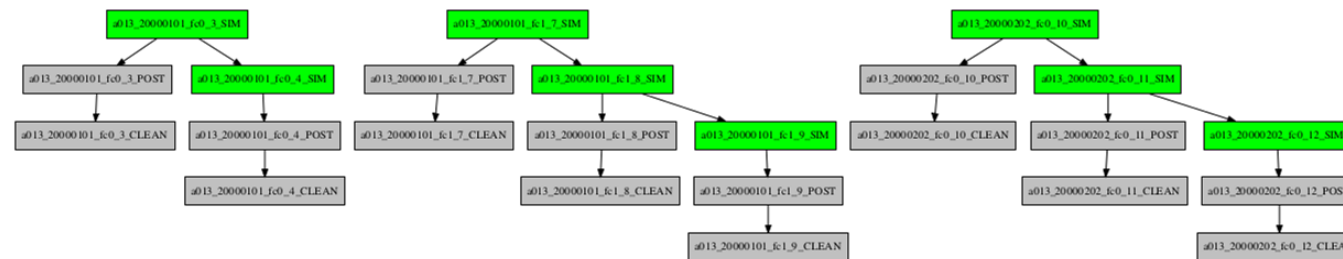
Autosubmit visualization

Grouping jobs by date, member, chunk, split; or automatically.



*Automatic behavior:
Collapsing jobs sharing status.*

*Hide groups:
Showing the more
relevant information.*



Autosubmit GUI

a142
NOT RUNNING

ORCA12-T1279 spin-up : a17z restart for the ocean, perpetual year 1950

Version: 3.9.0

Owner: 1946 tarsouze

Path: /esarchive/autosubmit/a142

SleepTime: 10

Last Modified: 2019-09-21 18:29:59

Last Access: 2019-10-13 16:47:27

Pkl: 1568162245

Model: https://earth.bsc.es/gitlab/es/auto-ecearth3.git

Branch: 3.2.2_Primavera_production_T1279-ORCA12

Hpc: marenostrum4

Show Tree View
Show Graph
Group By
Hide Log

Search by Job Name

Max children: 3 | Max parents: 2 | Total #Jobs: 3013

a142_19500101_fc0_602_SIM

Date: Sun, 01 Jan 1950 00:00:00 GMT

Section: SIM

Platform: marenostrum4

Processors: 5040 Wallclock: 07:00

Level: 605

Status: FAILED

Out: 3 In: 2

⬅️
⬆️
➡️

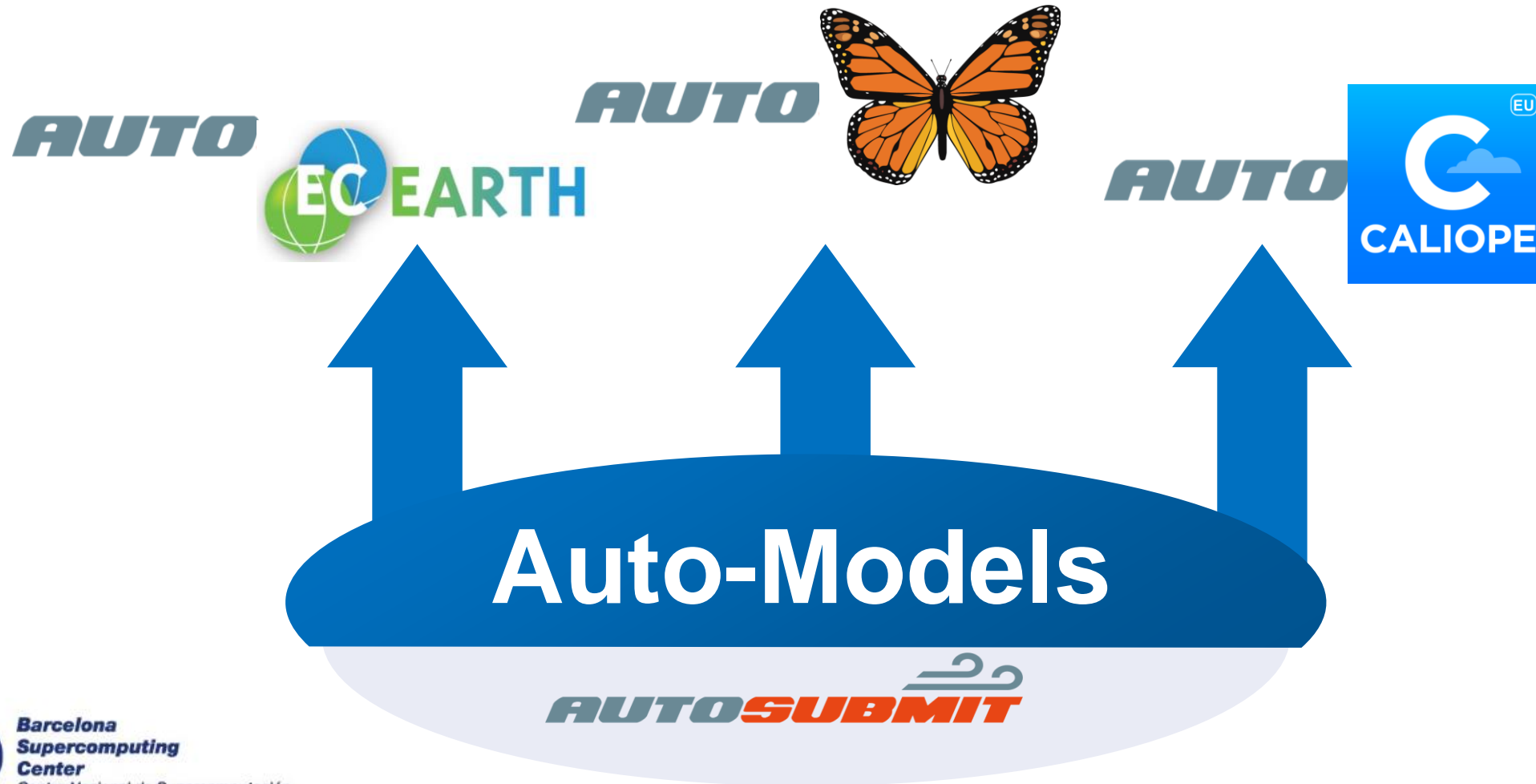
🏠
=
+

Waiting
Ready
Submitted
Queue
Running
Completed
Failed
Suspended
Unknown

Press the button so see the latest job with that status.

Auto-Models

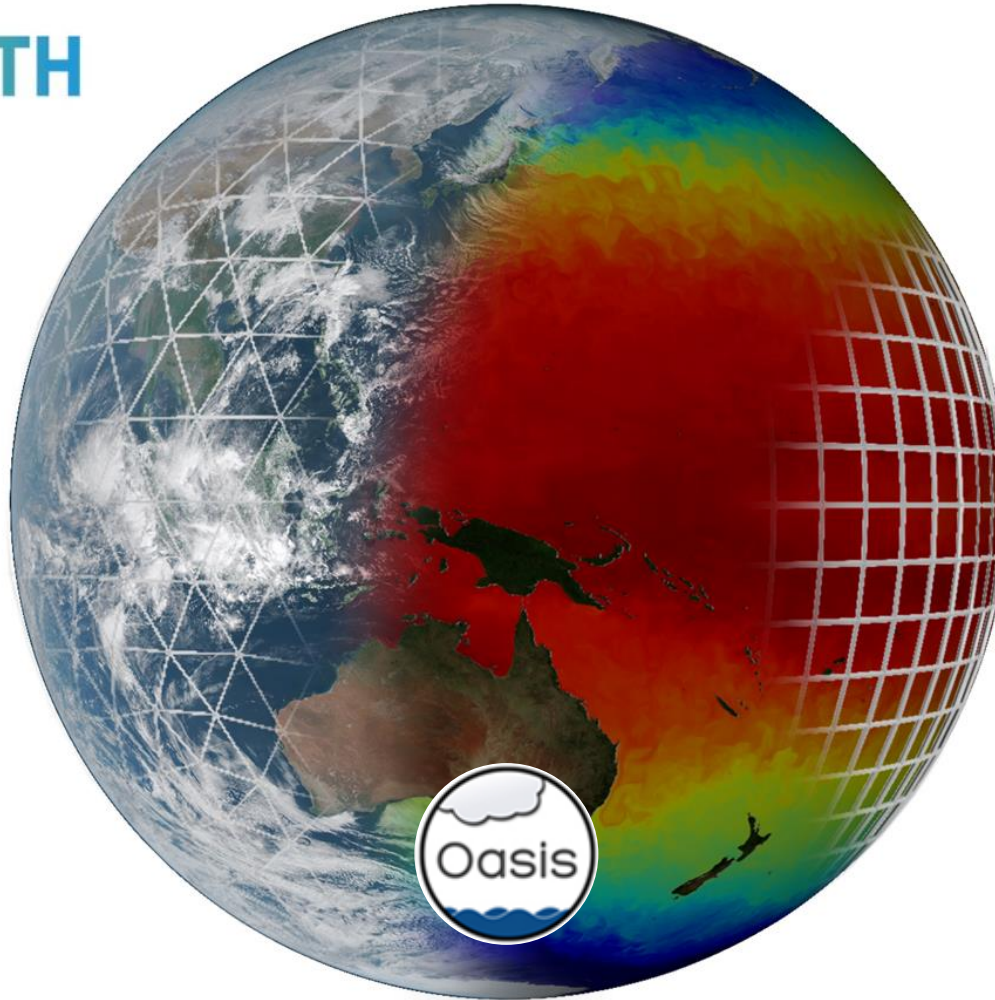
Autosubmit + Model source + Job templates + Auxiliary functions + Utilities



The EC-Earth ESM



Atmosphere:
IFS

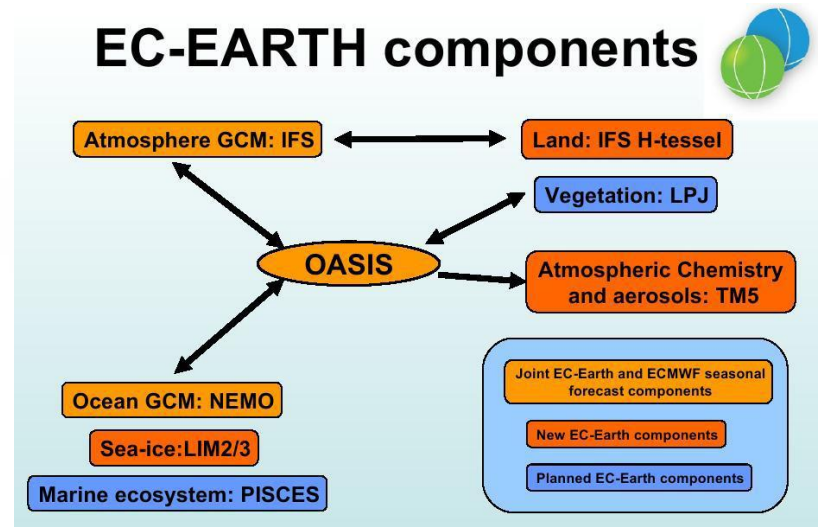


Ocean+ICE:

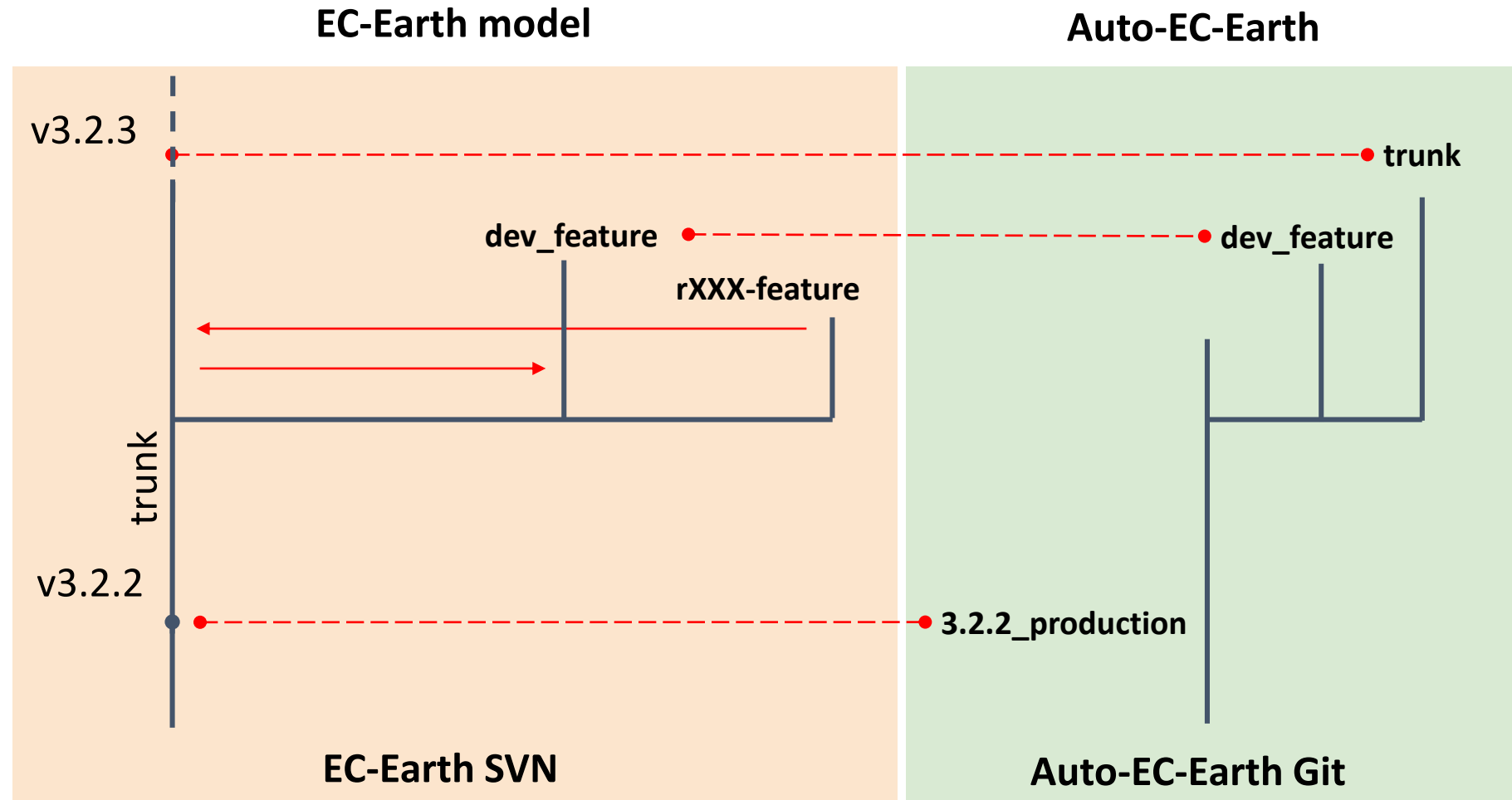


Coupler:

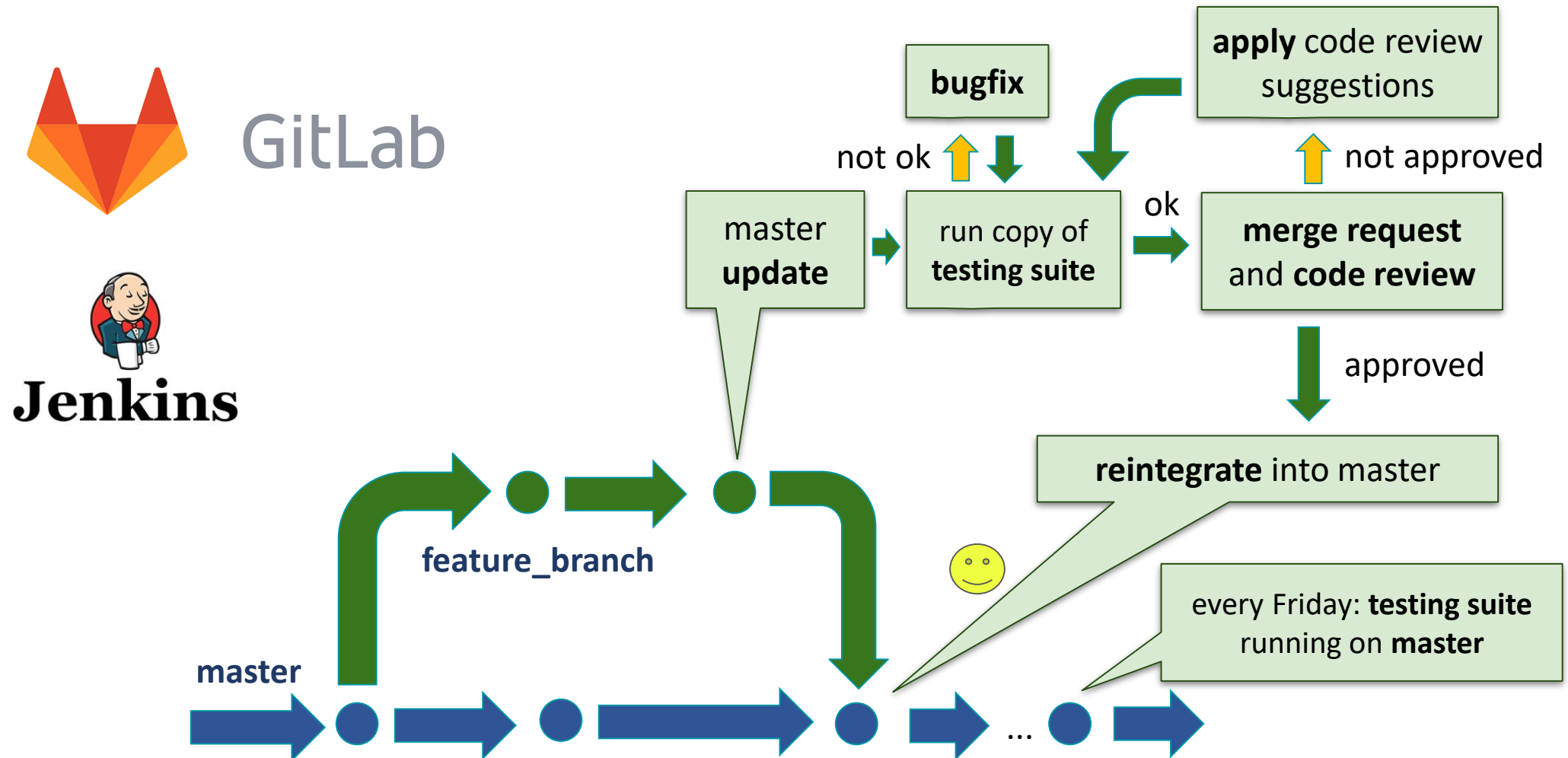
EC-EARTH components



Auto-EC-Earth branching



Auto-Models development



Auto-Models development

The screenshot displays a GitLab Kanban board with four columns: **Open** (76 items), **to do** (45 items), **working on** (22 items), and **testing** (4 items). Each column contains task cards with titles, status labels (e.g., 'stopped', 'bug', 'new feature'), and issue numbers. For example, in the 'Open' column, tasks include 'migrate experiment test' (#879), 'POST and CMOR failed a1k6 in trunk' (#769), and 'Add group permissions in /esarchive/exp/earth /EXPID/original_files/' (#756). The 'to do' column includes 'DT commands improvement' (#894) and 'DT logfiles should be deleted on successful operation' (#827). The 'working on' column features 'POST & CLEAN templates & Plugins refactory' (#936) and 'SYNCHRONIZATION - If project folder doesnt exist, override UPDATE_MODEL' (#931). The 'testing' column shows 'OASIS INI member' (#950) and 'Improve ECMWF_SYNCHRONIZATION' (#873).



GitLab

Merge requests

GitLab board – Agile “Kanban” methodology

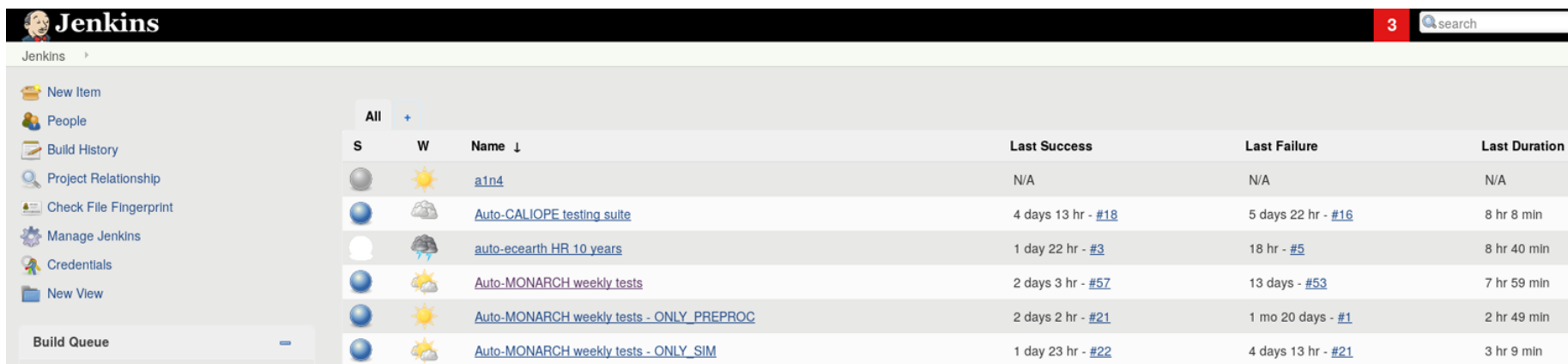
The screenshot shows a GitLab Merge Request interface. At the top, it says 'Request to merge 1017-development-CL into trunk' with buttons for 'Open in Web IDE', 'Check out branch', and a download icon. Below this, there's a 'Merge' button with checkboxes for 'Delete source branch' and 'Squash commits'. A summary line states '47 commits and 1 merge commit will be added to trunk.' Below that, it says 'You can merge this merge request manually using the command line'. At the bottom, there are thumbs up/down icons, a 'Discussion' section with 19 items, and a status bar showing 'Commits 47', 'Changes 12', 'Show all activity', and '6/6 threads resolved'.

Auto-EC-Earth testing: release tests

For every version release: run a complete set of tests. Every week: run a smaller set of tests.

nord3	CCA	MN4	resolution	type	details
t02c	t00u	t00q	T255L91-ORCA1L75-LIM3	coupled	start from restart
		t00v	T255L91-ORCA1L75-LIM3	coupled	atmos. nudging
		t011	T255L91-ORCA1L75-LIM3	coupled	sppt
	t00s	t00o	T255L91-ORCA1L75-LIM3	coupled	cold start
	t01d	t00z	ORCA1L75-LIM3	nemo	cold start
		t01j	ORCA1L75-LIM3	nemo	cold start ocean nudging
	t01e	t00r	T511L91-ORCA025L75-LIM3	coupled	start from restart
		t01o	ORCA025L75-LIM3	nemo	cold start
	t01b	t00y	T511L91	ifs	cold start
	t00t	t00p	T511L91-ORCA025L75-LIM3	coupled	cold start

Integration tests: testing suite



The screenshot shows the Jenkins web interface. On the left is a sidebar with navigation links: New Item, People, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, Credentials, and New View. Below these is a 'Build Queue' section. The main area displays a table of jobs. The table has columns for status (S), weather icon (W), name, last success, last failure, and last duration. A red badge with the number '3' is in the top right corner, indicating the number of failed builds.

S	W	Name ↓	Last Success	Last Failure	Last Duration
☐	☀️	a1n4	N/A	N/A	N/A
🌐	☁️	Auto-CALIOPE testing suite	4 days 13 hr - #18	5 days 22 hr - #16	8 hr 8 min
☐	☁️	auto-ecearth HR 10 years	1 day 22 hr - #3	18 hr - #5	8 hr 40 min
🌐	☀️	Auto-MONARCH weekly tests	2 days 3 hr - #57	13 days - #53	7 hr 59 min
🌐	☀️	Auto-MONARCH weekly tests - ONLY_PREPROC	2 days 2 hr - #21	1 mo 20 days - #1	2 hr 49 min
🌐	☀️	Auto-MONARCH weekly tests - ONLY_SIM	1 day 23 hr - #22	4 days 13 hr - #21	3 hr 9 min

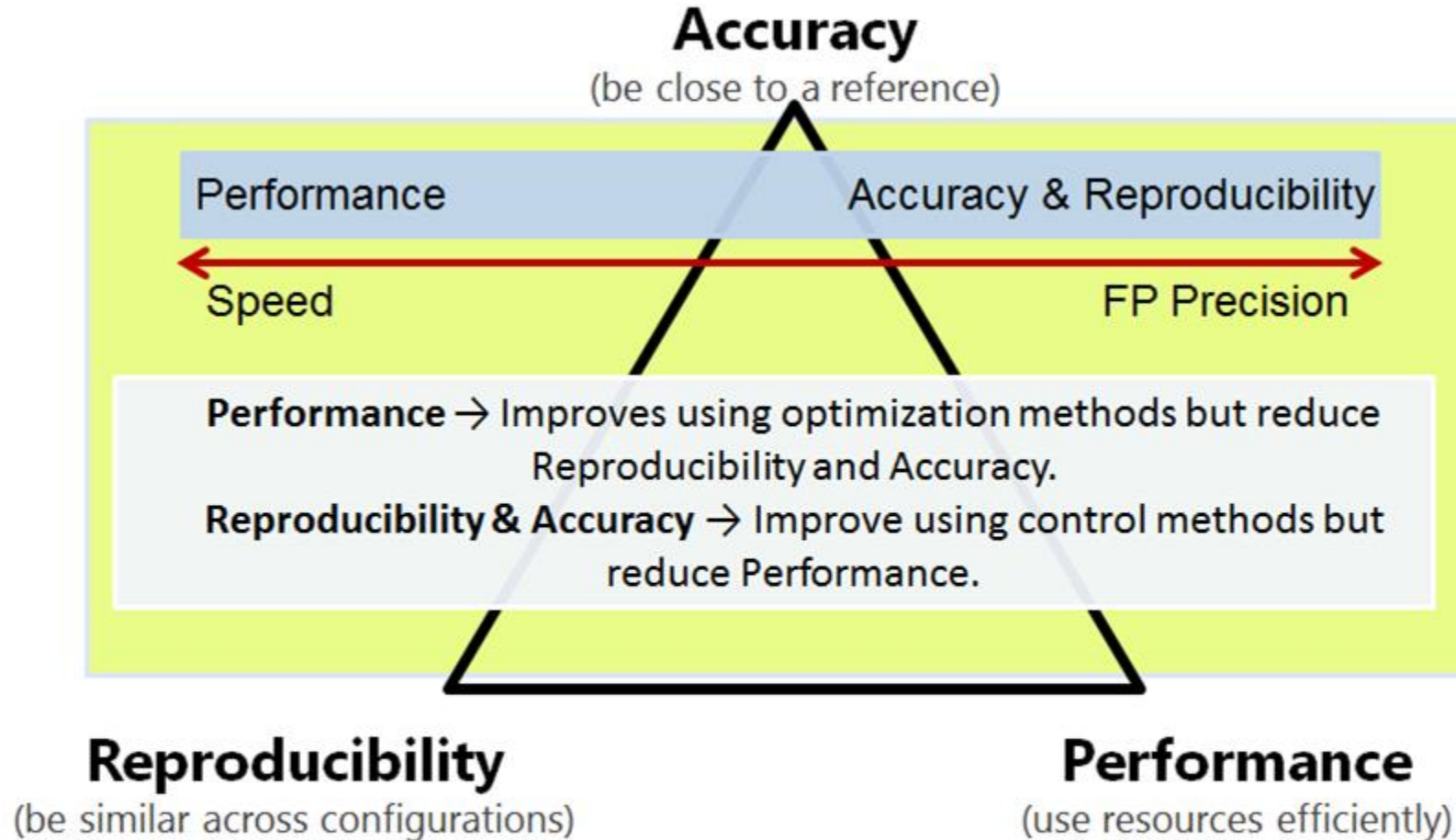


nord3	CCA	MN4	resolution	type	details
t02c	t00u	t00q	T255L91-ORCA1L75-LIM3	coupled	start from restart
		t00v	T255L91-ORCA1L75-LIM3	coupled	ATM nudgin
		t011	T255L91-ORCA1L75-LIM3	coupled	sppt
	t00s	t00o	T255L91-ORCA1L75-LIM3	coupled	cold start
	t01d	t00z	ORCA1L75-LIM3	nemo	cold start
		t01j	ORCA1L75-LIM3	nemo	cold start ocean nudging
	t01e	t00r	T511L91-ORCA025L75-LIM3	coupled	start from restart
		t01o	ORCA025L75-LIM3	nemo	cold start
	t01b	t00y	T511L91	ifs	cold start
	t00t	t00p	T511L91-ORCA025L75-LIM3	coupled	cold start

Reproducibility

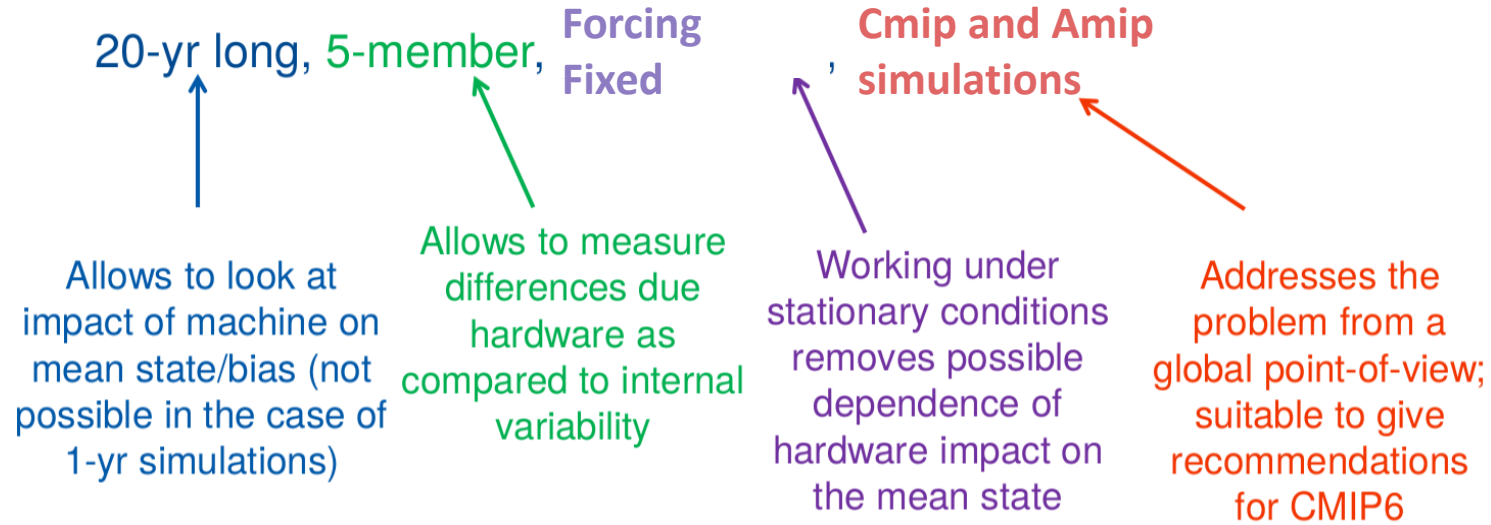
- Earth models: variety of **spatial resolutions, configurations** and running **environments**.
- Scientific codes are often in a **near-constant state of development** as new science capabilities are added and requirements change.
- If we want to study the model response under some scientific changes, we have to ensure that computational **changes** do not affect the **results**.
- We need a **method** to evaluate the **computational efficiency** of our models:
 - When the hardware changes, the number of resources or the configuration changes.

Reproducibility



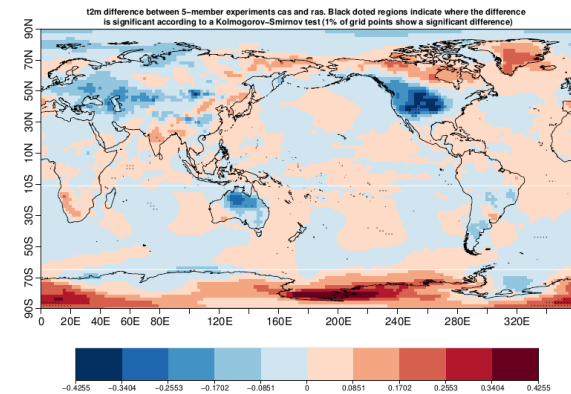
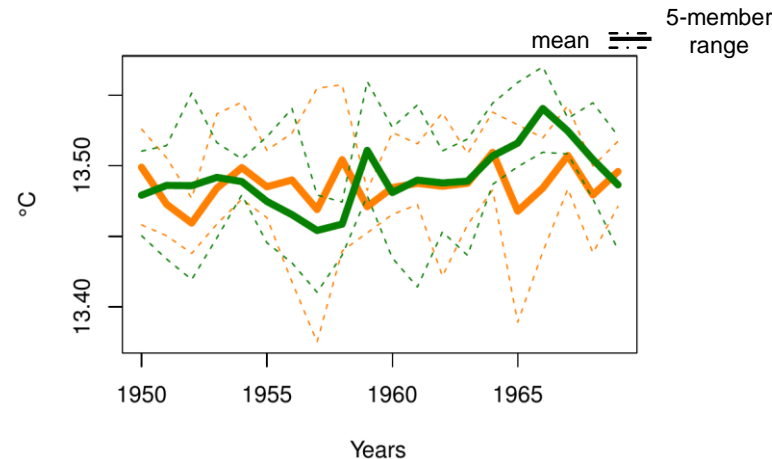
Find options to control the tradeoffs among accuracy, reproducibility and performance.

Methodology: EC-Earth CMIP6 case

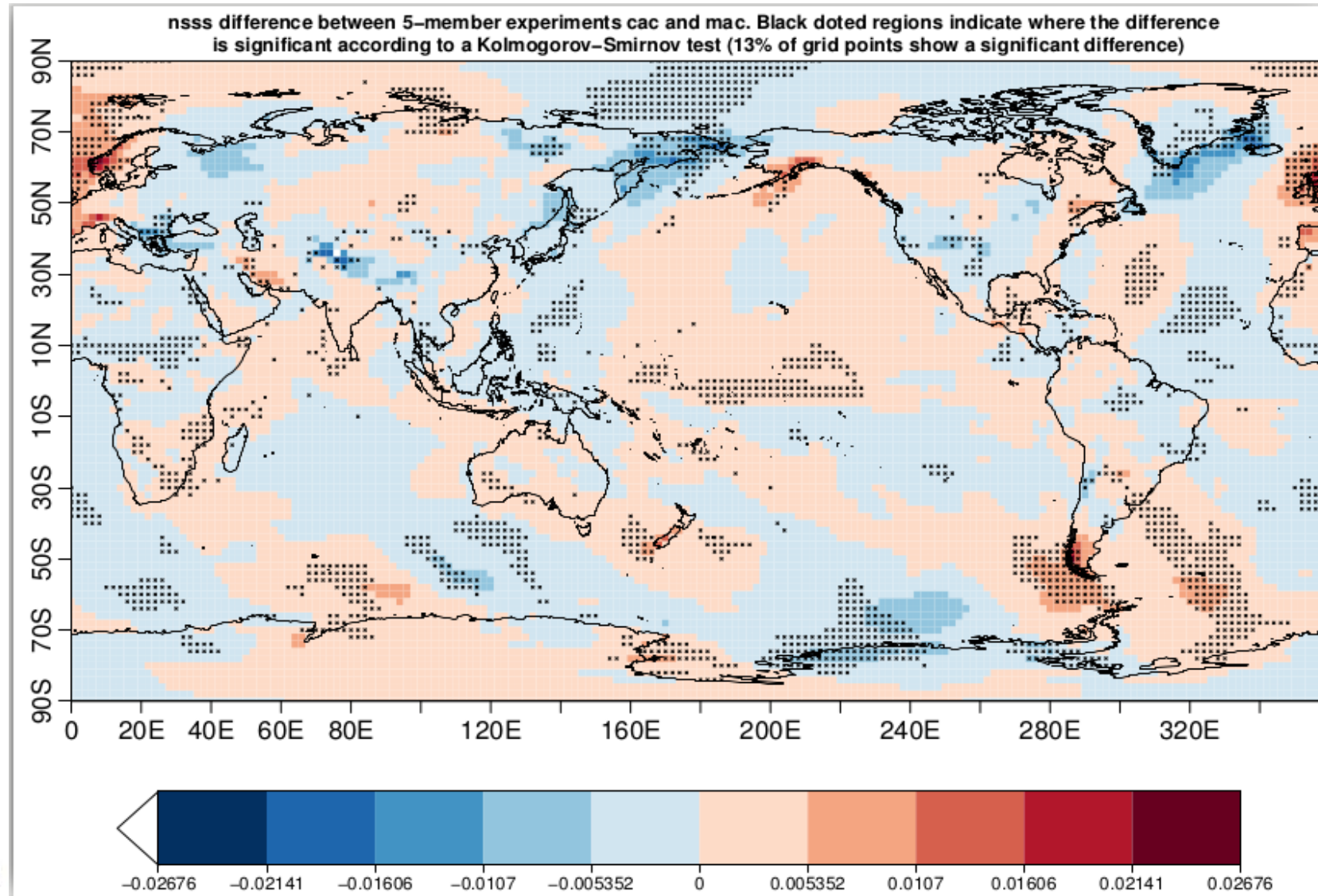


In comparison:

- Platforms
- Compilers
- Domain Decomposition
- Model options




Reproducibility: Kolmogorov-Smirnov



Methodology

Performance

- 
- **(no-prec)**-O3 -xHost -r8 -ipo -prof-use -no-prec-div -no-prec-sqrt
 - **(prof-use)**-O3 -xHost -r8 -ipo -prof-gen → -O3 -xHost -r8 -ipo -prof-use
 - **(ipo)**-O3 -xHost -r8 -ipo
 - **(O3)**-O3 -xHost -r8
 - **(O2)**-O2 -xHost -r8
 - **(no-fma_fz)**-O2 -no-fma -ftz -r8
 - **(fp-except)**-O2 -fp-model except -no-fma -ftz -fpe0 -r8
 - **(fp-precise)**-O2 -fp-model precise -fimf-arch-consistency=true -no-fma -fpe0 -r8
 - **(fp-strict)**-O2 -fp-model strict -fimf-arch-consistency=true -no-fma -fpe0 -r8
 - **(O1)**-O1 -fp-model strict -fimf-arch-consistency=true -no-fma -fpe0 -r8

Accuracy

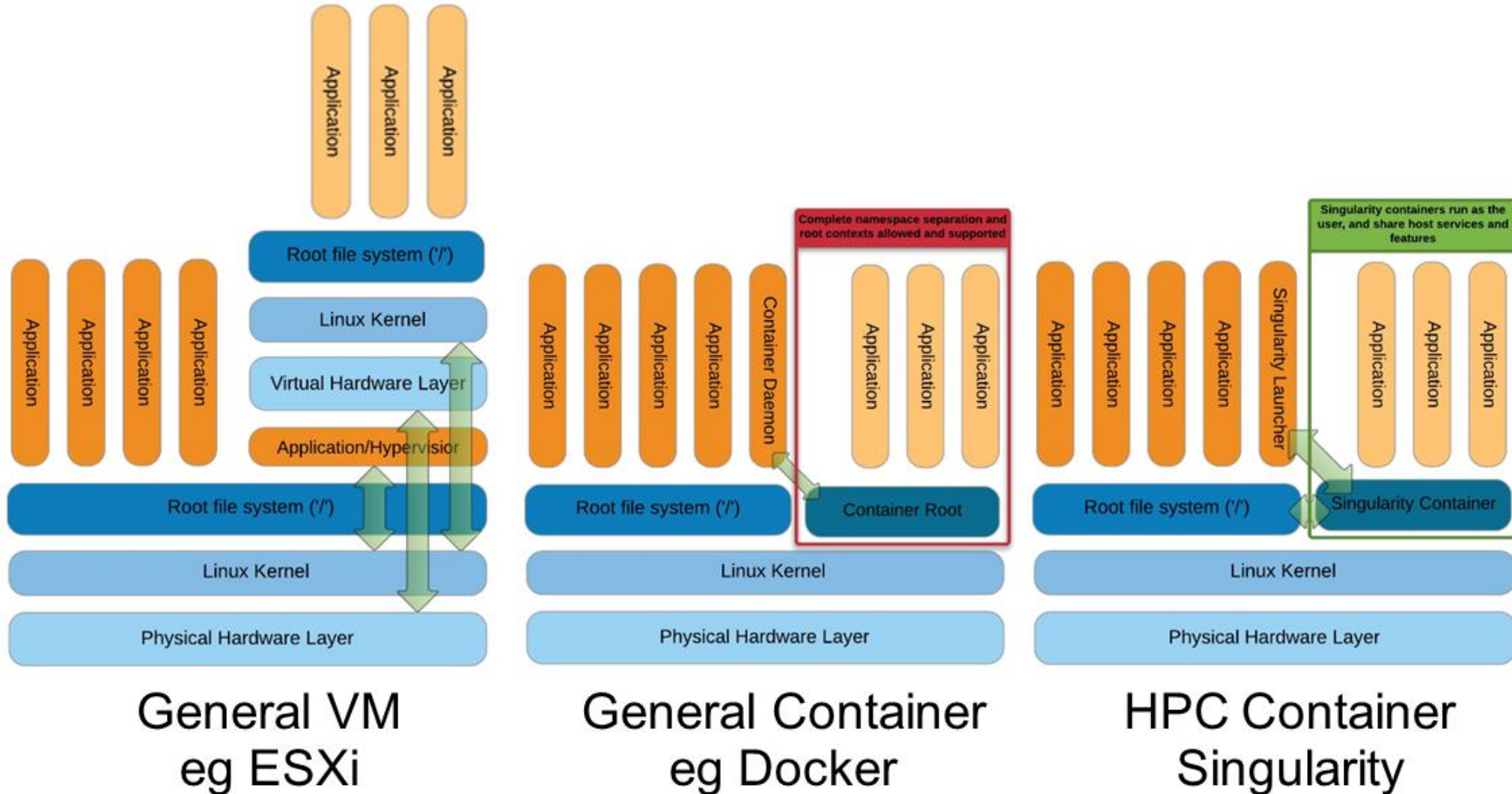
Conclusions: EC-Earth model

- **“Standard” flag** configurations for performance and FP precision obtain the **best results**.
 - -fp-model precise -fpe -no-fma -O2 -xHost -r8
 - Good performance, good precision, avoid fp errors, catch fpe exceptions.
 - -O2 -xHost -r8
 - Better performance (6%), good precision.
- **Aggressive optimizations** (O3, ipo, prof-use) **do not improve the performance**.
 - Some issues may avoid additional optimizations (loop dependences, non vectorization, MPI overhead ...).
- **Strict FP control does not improve the precision and reduce the performance** up to 6%-12%.
- Using approximations for FP operations (no-prec-div/sqrt) do not improve the performance and reduce the precision and reproducibility dramatically.

Containerisation

- **Common recurrent problem in scientific environments:** reproducibility between different environments.
- **Containers:** make possible the virtualization at the operating system level.

Containerisation



Containerisation

- **Proposed technologies:** Singularity and Shifter, both designed to be used in HPC environments:



- **Main questions:**
 - What to containerize? The whole workflow, just the model?
 - Analyze the impact of containers within the HPC environment.
¿Computational performance?

MPMD case with singularity - Challenge: isolation level

- **Isolate** the complete **environment** is **complex** (network setup, node visibility, compiler license, etc). Then:

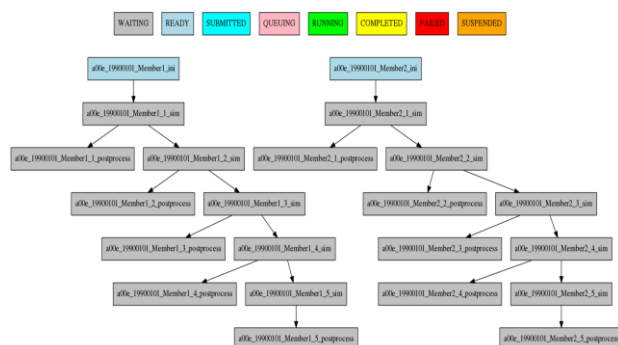
```
./mpirun -np N singularity exec container.img exe1 : \  
-np M singularity exec container.img exe2
```

- The executables need **binding** dynamic libraries folder and other system folders, so the call should be like:

```
mpirun -np N singularity --bind /apps:/apps /opt:/opt /lib64:/lib64 exec container.img exe1 : \  
-np M singularity --bind /apps:/apps /opt:/opt /lib64:/lib64 exec container.img exe2...
```


Performance metrics & performance reproducibility

Climate predictions have complex workflows.
New metrics are needed to evaluate
the computational efficiency. →



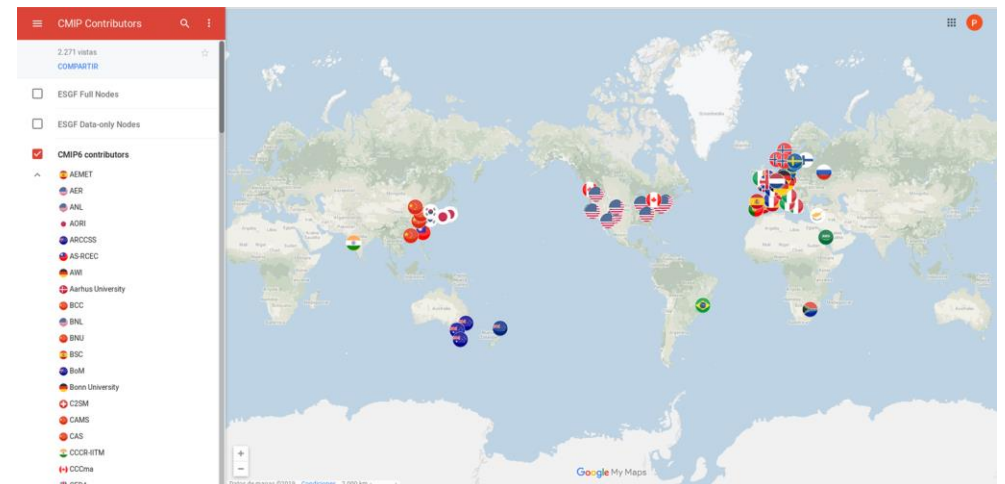
- The simulation does **not only** involve the execution of a model during a sequence of time steps (represented by the sim job).
- The experiment **adds complexity in the horizontal for ensembles** (members with perturbations in the initial conditions).
- The experiment **adds complexity in the vertical, running long simulations** divided into chunks and including **pre- and post-processing**.

Metric	used to evaluate ...
Parallelization	Number of parallel resources allocated
Simulation Year Per Day (SYPD)	how efficient is your run job per each year of the simulation
Core-hours Per Year (CHPY)	how efficient is your run job with respect to the number of parallel resources used
Run Time (Average and Total)	how much time are per job and all the jobs in the critical path running
Queue Time (Average and Total)	how much time are per job and all the jobs in the critical path waiting in the queue
Post Run T. (Average and Total)	how much time are per job and all the jobs in the critical path running the post
Post Queue T. (Average and Total)	how much time are per job and all the jobs in the critical path waiting in the queue the post
Actual SYPD	how affect queue time to the complete experiment, from the first to the last run job
Real SYPD	How affect other issues (quota, jobs failing, rerun...) the time to complete an experiment
Energy Cost Per Year (JCPY)	how much energy is needed per each year of simulation
Data Volume	Ouput produced per year Simulated
Load Balance Cost	how much time is wasted in waiting time, produced by irregular executions

- The execution of a coupled model is complex. **Different components run in parallel**, exchanging some information and adding some extra overhead (Communication and Interpolation Time). **Load imbalances produce extra waiting time as well**.
- The execution of **each component could involve some irregular extra overhead**, such as output processing or calculations occasionally done, such as radiation.

ESGF

- The **Earth System Grid Federation (ESGF)**: international collaboration serving the World Climate Research Programme's (WCRP) Coupled Model Intercomparison Project (CMIP) and supporting climate and environmental science in general.
- Web **interface** to the majority of experimental datasets used in model intercomparison projects.
- Data published have **PIDs** and **DOI**, and are heavily quality controlled.
- Data can be **discovered** and accessed through THREDDS.
- Tenths of PB of “Earth data” available.



Some current issues

- Initial **data version control**: Need of a centralised and accessible system to share model initial data. Facilitate reproducibility and save space by incremental diff.
- Containers: Difficult to **isolate HPC environment**. Is it needed? Is it enough with statically compiled binaries?
- Model workflows are strongly based on data movement. Not so straightforward to implement **unit tests**.
- **Science reproducibility** (not b2b) still expert **human** intervention.



**Barcelona
Supercomputing
Center**
Centro Nacional de Supercomputación



Thank you



esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER
AND CLIMATE IN EUROPE

The ESiWACE project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 675191

miguel.castrillo@bsc.es

Methodology: Reproducibility Test

- Accuracy and reproducibility

- Reichler-Kim normalized index

- Calculate for each variable a normalized error variance e^2 by squaring the grid-point differences between simulated and observed climate

$$e_{vm}^2 = \sum_n \left(w_n (\bar{s}_{vmn} - \bar{o}_{vn})^2 / \sigma_{vn}^2 \right)$$

- Ensure that different climate variables receive similar weights when combining their errors

$$I_{vm}^2 = e_{vm}^2 / \overline{e_{vm}^2}^{m=20C3M}$$

- Mean over all climate variables

$$I_m^2 = \overline{I_{vm}^2}^v$$

- Kolmogorov-Smirnov (KS) test

- Compare two five-member ensembles and determine whether the two ensembles are statistically indistinguishable from one another.
 - Since no prior assumption can be made on the underlying statistical distribution of the samples, the non-parametric KS test is suitable for the evaluation.

Methodology: Reproducibility Test

- About the reproducibility methodology
 - F. Massonnet, M. Ménégos, M.C. Acosta, X. Yepes-Arbós , E. Exarchou, F.J. Doblas-Reyes
 - https://earth.bsc.es/wiki/lib/exe/fetch.php?media=library:external:technical_memoranda:massonnet_etal_bsc-techmem-reproducibility2018.pdf
- About the reproducibility test for EC-Earth
 - Adapted by Philippe Le Sager for EC-Earth community
 - <https://github.com/plesager/ece3-postproc#reproducibility-test>
- About the reproducibility results and how to collaborate
 - Contact Mario (mario.acosta@bsc.es) and Philippe (sager@knmi.nl)
 - Provide feedback: <https://dev.ec-earth.org/issues/533>
 - People collaborating: Uwe Fladrich (SMHI), Jose M. Rodríguez (AEMET), Paul Nolan (ICHEC)

Methodology: CPMIP Metrics

- About the CPMIP Metrics
 - V. Balaji et al. 2017
 - <https://www.geosci-model-dev.net/10/19/2017/gmd-10-19-2017.pdf>
- About the adaptation for EC-Earth
 - Adapted by Uwe Fladrich and Domingo Manubens for EC-Earth
 - Adapted by Philippe Le Sager for common CCA (ECMWF) experiments
 - Adapted and extended by Mario Acosta and Domingo Manubens for MN4 (BSC) using a workflow manager
- About future plans
 - Improvement and Portability of CPMIP metrics for ESMs in IS-ENES3 (Mario Acosta, Eric Maissonaive...).
 - Work in collaboration with the community (V. Balaji, Pier-Luigi...)
- About the CPMIP results and how to collaborate for CMIP6 runs
 - Contact Mario (mario.acosta@bsc.es) and Philippe (sager@knmi.nl)
 - Provide feedback: <https://dev.ec-earth.org/issues/532>
 - People collaborating: Uwe Fladrich (SMHI), Jose M. Rodríguez (AEMET), Paul Nolan (ICHEC)

- What it is:
 - <https://esgf-node.llnl.gov/search/esgf-llnl/>
 - web portal to download climate (CMIP5-6, SPECS, obs4MIPS, CORDEX,...) data
 - ingests very standardized data (CMOR-\$proj)
- There are data nodes and index nodes:
https://www.google.com/maps/d/u/0/viewer?mid=1qgltlbd11j6wzirf6vo2Zq_uYqaP4baQ&ll=43.83402373129155%2C14.074149978563128&z=4
 - Data nodes **host the data** and have a thredds server displaying the data hosted locally
 - <https://esgf.bsc.es/thredds/catalog/catalog.html>
 - Indexes have a landing page allowing to search